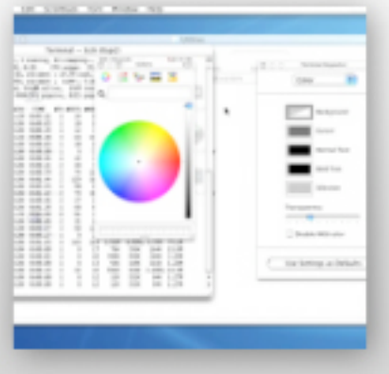
Writing Threaded Apps on Mac OS X

Session 111



Writing Threaded Apps on Mac OS X

Mark Tozer-Vilchez
Desktop Hardware Evangelist
tozer@apple.com

Writing Threaded Apps on Mac OS X

George Warner
Schizophrenic Optimization Scientists
geowar@apple.com

Agenda

- Threading terminology
- Why?
- Why not?
- Threading architectures
- Do’s and Don’ts
- Demo’s
- Q&A

Threading Terminology

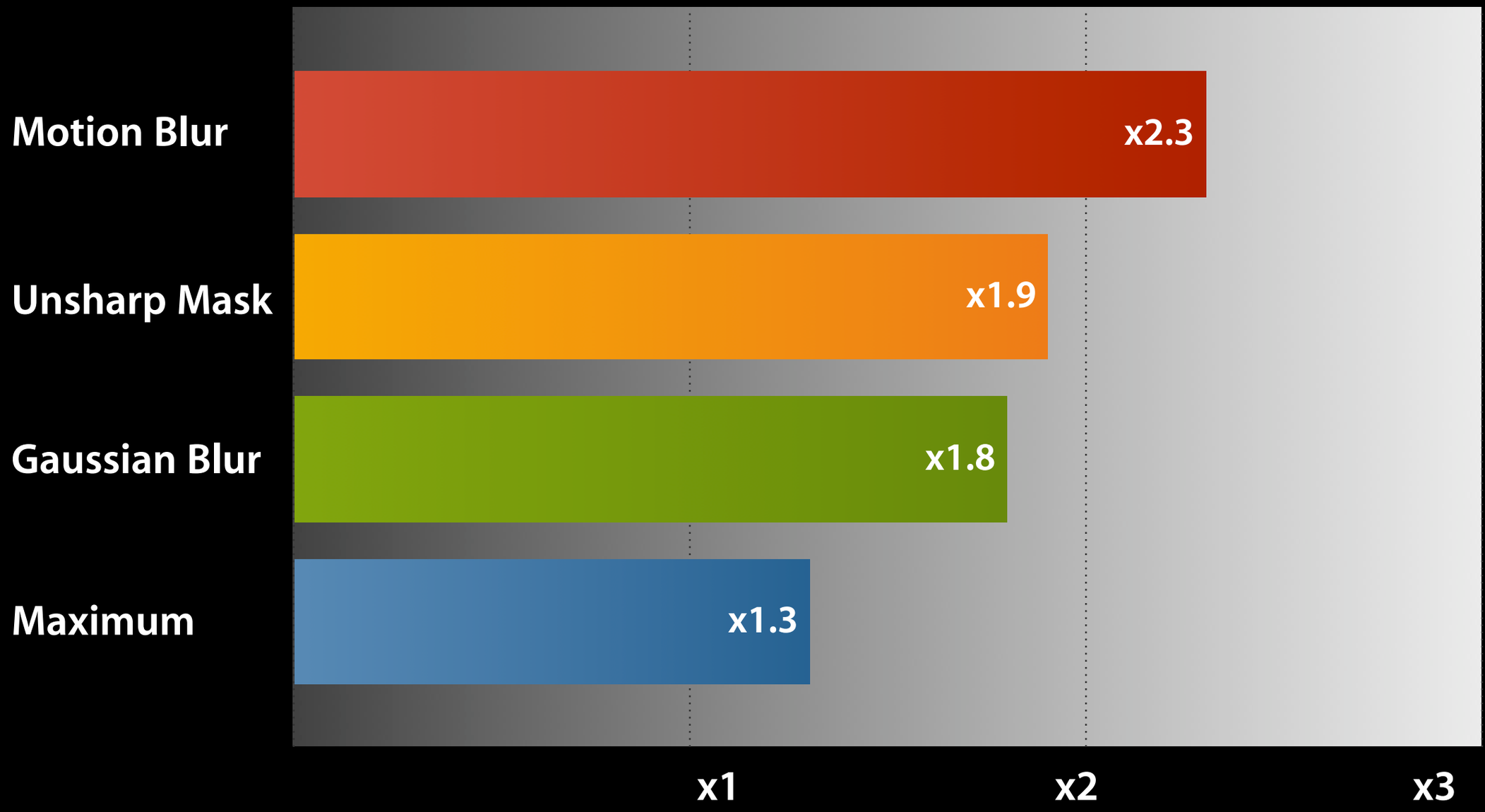
- Threads vs. processes
- A thread is a independent execution path preemptively or cooperatively scheduled
- A process is a collection of threads plus the resources necessary to run those threads
- Multiprocessing is a general case of multitasking container that holds a set of threads

Why Use Threads?

- Customer expectations
  - Responsive GUI
- Scalability
  - MP Hardware equals MP Performance!

Scalability

Wider is better!



Why Use Threads?

- Preemption
  - More threads equals more CPU time
- Synchronous requests
  - Put blocking calls into their own thread
- Polling is bad
  - Block Instead

When to Avoid Threads

- Added complexity
  - Global data requires locks
  - Shared data may require locks
  - Non Thread-safe APIs

When to Avoid Threads

- Added overhead
  - Creation time
    - 300 µSec (1 GHz Dual G4)
  - Preemption time
    - 40 µSec (1 GHz Dual G4)
  - Memory footprint
    - 512 KByte default (virtual) stack

When to Avoid Threads

- Kernel resources
  - Memory footprint
    - Hardware context storage

When to Avoid Threads

- Kernel resources
  - Memory footprint (≈ 2K per thread)
    - Hardware context storage
      - 32 General-Purpose Registers
      - 32 Floating-Point Registers
      - FP Status & Control Register
      - XER, Link & Count Registers
      - Memory Management Registers
      - Velocity Engine Registers
- 100s of threads

When to Avoid Threads

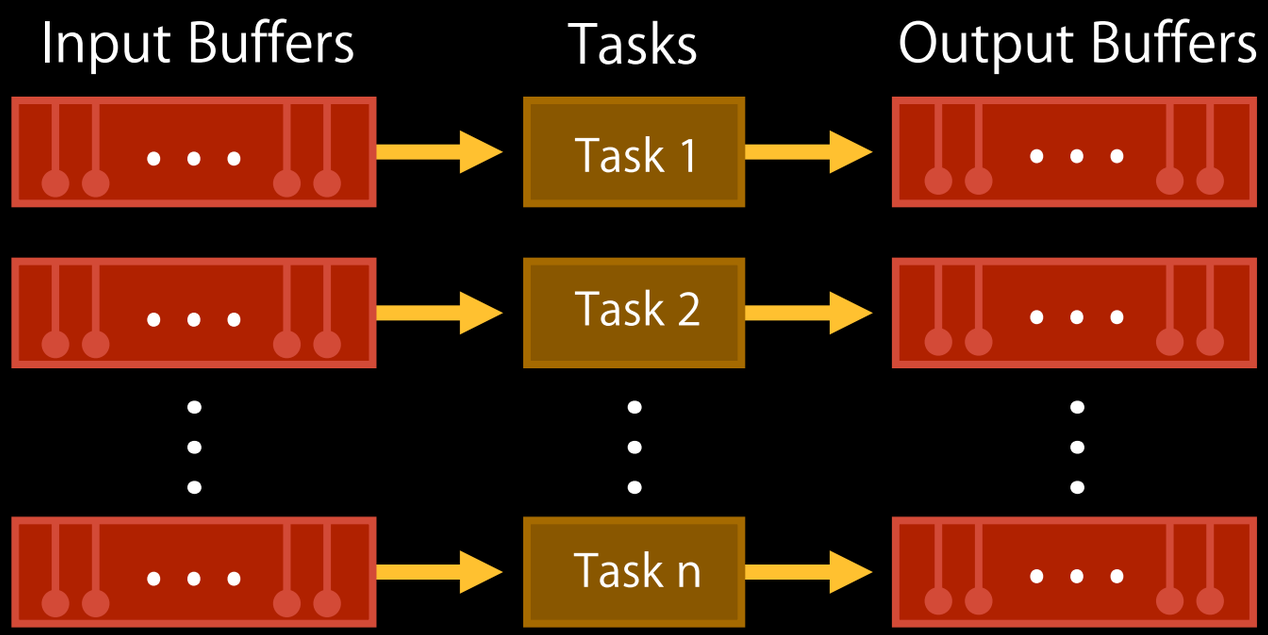
- Other options
  - Cooperative threads
  - Timers

Threading Architectures

- Parallel Tasks with Parallel I/O Buffers
- Parallel Tasks with Shared I/O Buffers
- Sequential Tasks with Multiple I/O Buffers

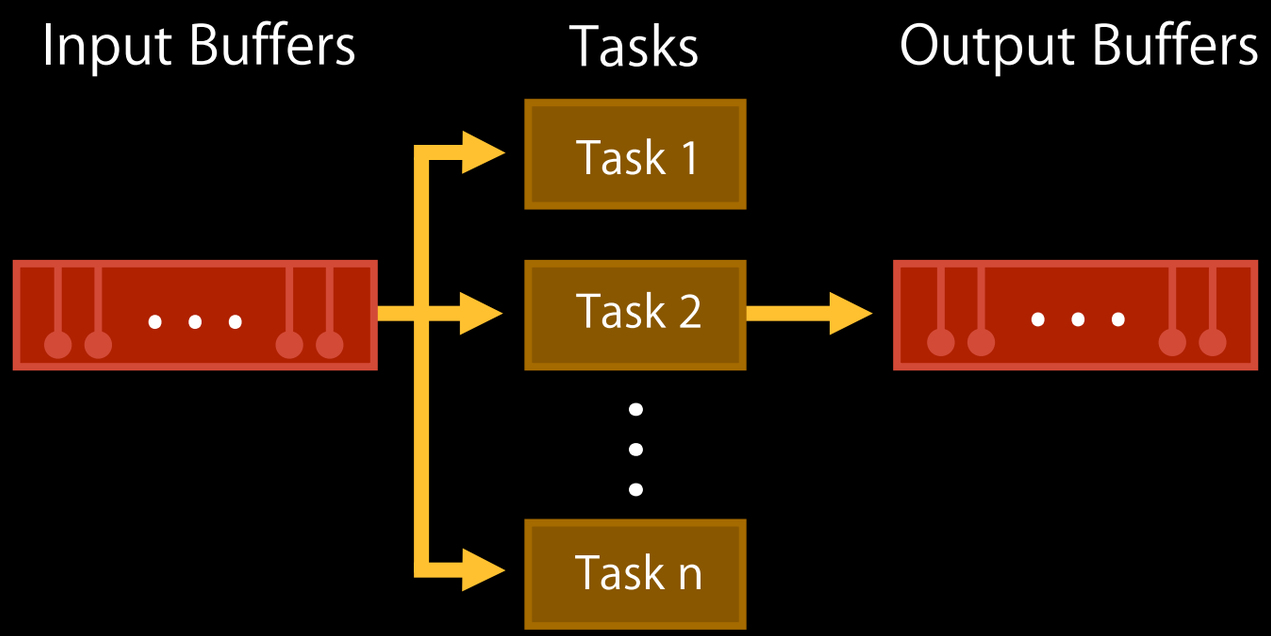
Threading Architectures

Parallel Tasks with Parallel I/O Buffers



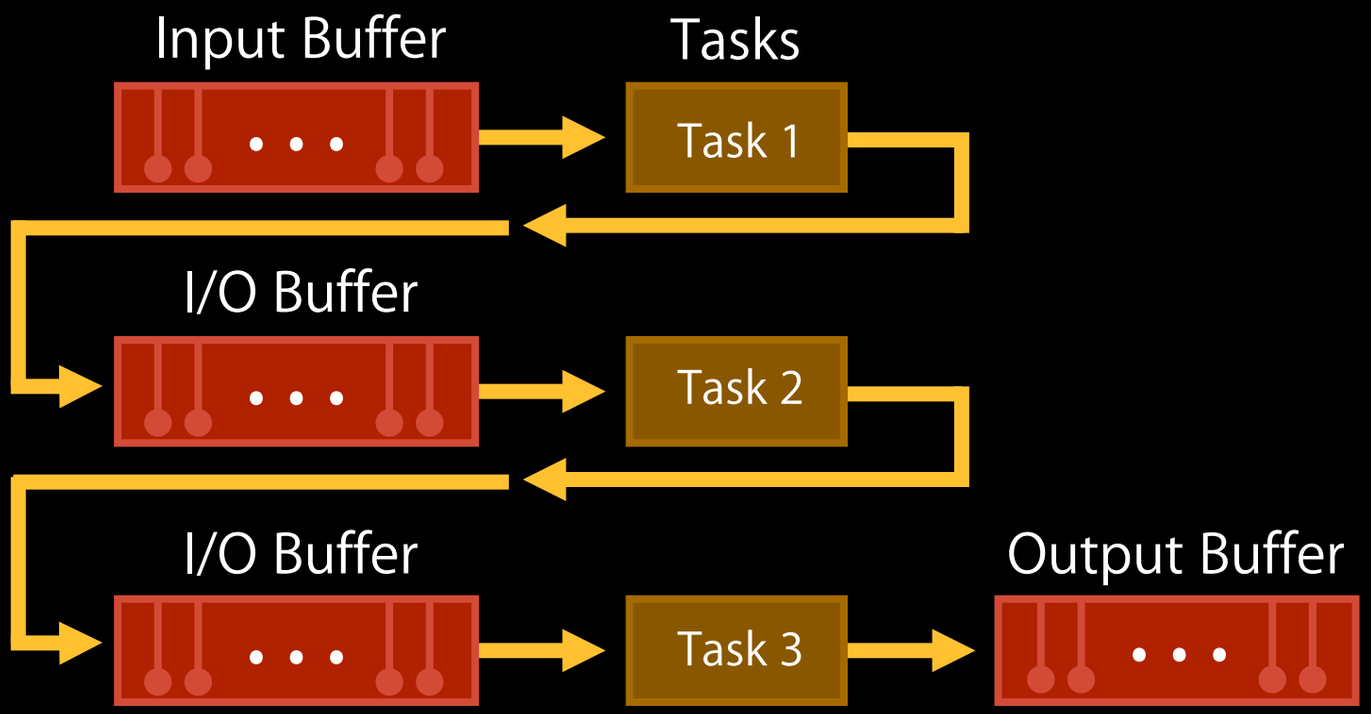
Threading Architectures

Parallel Tasks with Shared I/O Buffers



Threading Architectures

Sequential Tasks with Multiple I/O Buffers



Threading Architectures

- Many applications have parallel and independent execution paths
- Examples:
  - A word processor
  - A driving simulator
  - An image processor

Threading Implementations

Java
MP API

Carbon
NS Threads

Cocoa
Java Threads

Darwin
pthreads

Threading Implementations

- Common Concepts
  - Thread management
  - Synchronization primitives
  - Thread-safe services

Do’s and Don’ts

- Avoid Creation and Destruction Overhead
  - Pre-allocate and Use Pools

Do's and Don'ts

- Do be Data-driven
  - Prologue (Open, Allocate, etc.)
  - CRUNCH!
  - Epilogue (Close, Dispose, etc.)

Do’s and Don’ts

- Don’t Suspend, Resume, or Terminate
  - Use Synchronization Methods Instead

Do’s and Don’ts

- Don’t Under or Over Lock
  - Find a Balance

Do’s and Don’ts

- Avoid Spin Waiting
  - Use Separate Threads to Merge Signals

Do’s and Don’ts

- Don’t GUI
  - Quartz OK
  - OpenGL OK
  - Use PostEventToQueue

Demo

George Warner
geowar@apple.com

http://developer.apple.com/wwdc2003/urls.html

Wrap Up

Roadmap

100 IOKit: Kernel Extension Programming Techniques

Techniques for Writing Kernel Extensions

Russian Hill
Mon., 3:30pm

Who to Contact

George Warner
Senior Support Scientist
geowar@apple.com

Mark Tozer-Vilchez
Desktop Hardware Evangelist
tozer@apple.com

http://developer.apple.com/wwdc2003/urls.html

For More Information

Documentation

Threading

Carbon Threads
Multiprocessing Services Reference
ADC Home > Documentation > Carbon > Performance >

Cocoa Threads
Introduction to Multithreading
ADC Home > Documentation > Cocoa > Performance >

More Documentation

Threading

- POSIX Threads
  - man pages
  - Darwin CVS repository
    - Libc
    - xnu

http://www.opengroup.org

news://comp.programming.threads

Reference Library

Documentation

DiscRecording Services

ADC Home > User Experience > DiscRecording Services

Game Sprockets Reference

ADC Home > Documentation > Legacy Technologies >

Sound Manager

ADC Home > Documentation > Music & Audio >

Speech Technologies

ADC Home > User Experience > Speech Technologies

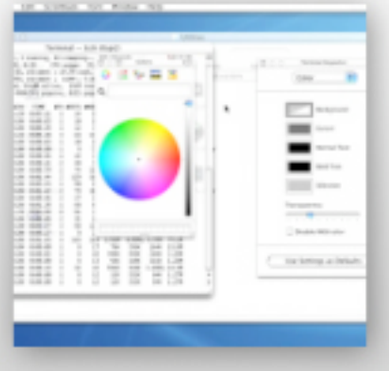
More Documentation

Threading

Technical Note 2028: Threading Architectures
http://developer.apple.com/technotes/tn/tn2028.html

Technical Note 2006: MP-Safe Routines
http://developer.apple.com/technotes/tn/tn2006.html

Q&A



George Warner
Schizophrenic Optimization Scientists
geowar@apple.com

http://developer.apple.com/wwdc2003/urls.html

WWDC 2003

Worldwide Developers Conference